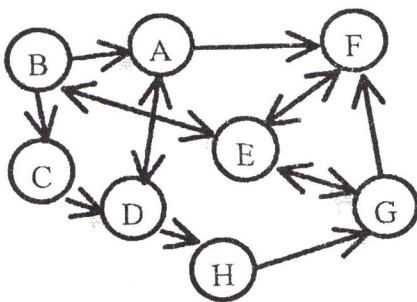


考試科目	資料結構及演算法	系別	資訊科學系	考試時間	2 月 24 日 (日) 第一節
------	----------	----	-------	------	------------------

- [4 pts] What are the definitions of the notations O (Big O) and Ω (Big Omega)?
- [5 pts] Here are 10 integers: 12, 10, 1, 6, 5, 4, 3, 2, 9, 7. Sort them using Merge sort and describe each step in detail.
- [16 pts; 4 pts for each] Describe time complexities (in the worst and average cases) of the following sorting algorithms. In addition, what kind of situations would you use these algorithms other than the others?
 - Quick sort.
 - Merge sort.
 - Radix sort.
 - Bucket sort.
- [6 pts; 3 pts for each] Given the input (14, 30, 99, 77, 7, 3, 21):
 - Construct a binary search tree according to the input sequence.
 - Delete the node "3".
- [6 pts] Insert a sequence of keys (30, 29, 28, 27, 26, 1, 20, 2, 19) into an empty AVL-tree. Show the AVL-tree step by step.
- [6 pts] Build a Huffman tree according to the following table, in which each character has its own frequency and give the corresponding Huffman codes for each character. (Note that 0 stands for the left direction and 1 for right direction.)

Character	A	E	I	O	S	Z	T
Frequency	10	15	12	13	3	1	4

- [7 pts] Use adjacency list and adjacency matrix to represent the graph and show the results of Breadth First Search (BFS) and Depth First Search (DFS) starting from node B.



考 試 科 目	資料結構及演算法	系 別	資訊科學系	考 試 時 間	2 月 24 日 (日) 第 一 節
---------	----------	-----	-------	---------	--------------------

8. [10 pts] Implement a queue with two stacks (**s1** and **s2**). You can use the following basic stack operations:
- ```
Data top(Stack s);
void push(Data d, Stack s);
void pop(Stack s);
int isEmpty(Stack s)
```
- Write the pseudo codes for the following three queue operations:
- Data first(Queue q);**
  - void enqueue(Data d, Queue q);**
  - void dequeue(Queue q);**
- (Note: You can use **q.s1** and **q.s2** to access the stacks **s1** and **s2** in the queue **q**.)
9. [10 pts] We call a boolean function  $f: \{0,1\}^k \rightarrow \{0,1\}$  *symmetric* if  $f(x_1, x_2, \dots, x_k)$  depends only on  $\sum_{i=1}^k x_i$ . How many symmetric boolean functions of  $k$  variables are there?
10. [10 pts] Prove that the halting problem is not decidable.
11. [10 pts] Let  $A[1 \dots n]$  be an array of  $n$  distinct numbers. If  $i < j$  and  $A[i] > A[j]$ , then the pair  $(i, j)$  is called an *inversion* of  $A$ . Suppose that the elements of  $A$  form a uniform random permutation of  $\langle 1, 2, \dots, n \rangle$ . Use indicator random variables to compute the expected number of inversions.
12. [10 pts] Show how quicksort can be made to run in  $O(n \lg n)$  time in the worst case, assuming that all elements are distinct.

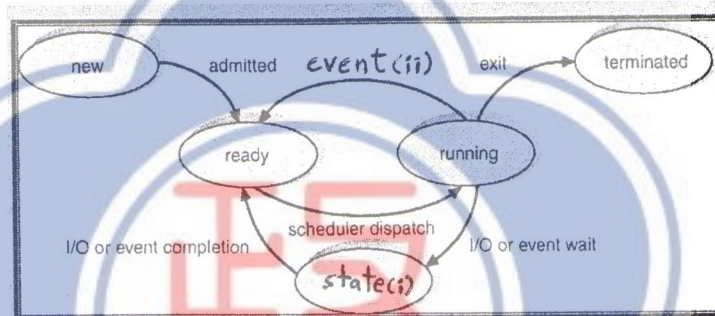
|      |      |    |       |      |              |
|------|------|----|-------|------|--------------|
| 考試科目 | 作業系統 | 系別 | 資訊科學系 | 考試時間 | 2月24日(日) 第二節 |
|------|------|----|-------|------|--------------|

I. Basic Concepts

- (5%) (a) What is the main advantage for an operating-system designer of using a virtual-machine architecture? What is the main advantage for a user?
- (5%) (b) What is the relationship between a guest operating system and a host operating system in a system like VMware? What factors need to be considered in choosing the host operating system?

II. Process Management

- (4%) (a) The process state transition diagram is shown as the following. Please fill in the blanks of state (i) and event (ii) in the diagram?



- (5%) (b) The producer-consumer problem (or bounded buffer problem) in the following code has fatal race condition, please explain how this condition happens?

```

Code for producer process
while (true) {
 /* produce an item in nextProduced */
 while (counter == BUFFER_SIZE)
 ; /* do nothing */
 buffer[in] = nextProduced;
 in = (in + 1) % BUFFER_SIZE;
 counter++;
}

```

```

Code for consumer process
while (true) {
 while (counter == 0)
 ; /* do nothing */
 nextConsumed = buffer[out];
 put = (out + 1) % BUFFER_SIZE;
 counter--;
 /* consumer the item in nextConsumed */
}

```

- (6%)(c) Please use semaphores to solve the problem (b) so that the fatal race condition cannot occur.
- (5%)(d) Why we have to guard against *race condition* and synchronize the *critical section*?

|    |        |
|----|--------|
| 備註 | 試題隨卷繳交 |
|----|--------|

|      |      |    |       |      |              |
|------|------|----|-------|------|--------------|
| 考試科目 | 作業系統 | 系別 | 資訊科學系 | 考試時間 | 2月24日(日) 第二節 |
|------|------|----|-------|------|--------------|

### III Memory Management

- (4%)(a) Under what circumstances do page faults occur? Describe the actions taken by the operating system when a page fault occurs.
- (6%)(b) What is the cause of thrashing? How does the system detect thrashing? Once it detects thrashing, what can the system do to eliminate the problem?
- (5%)(c) What is the Belady's anomaly? Why the stack algorithms, such as optimal replacement, least-recently-used (LRU) do not suffer from Belady's anomaly?
- (5%)(d) Consider a demand-paging system with a paging disk that has an average access and transfer time of 10 milliseconds. Addresses are translated through a page table in main memory, with an access time of 2 microsecond per memory access. Thus, each memory reference through the page table takes two accesses. To improve this time, we have added an associative memory that reduces access time to one memory reference, if the page-table entry is in the associative memory. Assume that 90 percent of the accesses are in the associative memory, and that, of the remaining, 10 percent (or 1 percent of the total) cause page faults. What is the effective memory access time?

### IV Storage Management

- (6%)(a) Consider a system that supports the strategies of *contiguous*, *linked*, and *indexed* allocation. What criteria should be used in deciding which strategy is best utilized for a particular file?
- (5%)(b) When opening and accessing a file, what is the difference between *memory-mapped I/O* approach and I/O using *read()* and *write()* approach?
- (4%)(c) The term *fast wide SCSI-II* denotes a SCSI bus that operates at a data rate 20 MB per second when it moves a packet of bytes between the host and a device. Suppose that a fast wide SCSI-II disk drive spins at 7200 RPM, has a sector size of 1024 bytes, and holds 160 sectors per track. Estimate the sustained transfer rate of this drive in megabytes per second?
- (5%)(d) Suppose that the drive has 7,000 cylinders, 20 tracks per cylinder, a head-switch time (from one platter to another) of 1 milliseconds, and an adjacent-cylinder seek time of 2 milliseconds. Use this additional information to give an accurate estimate of the sustained transfer rate for a huge transfer.

|    |        |
|----|--------|
| 備註 | 試題隨卷繳交 |
|----|--------|

|         |      |     |       |         |                  |
|---------|------|-----|-------|---------|------------------|
| 考 試 科 目 | 作業系統 | 系 別 | 資訊科學系 | 考 試 時 間 | 2 月 24 日(日) 第二 節 |
|---------|------|-----|-------|---------|------------------|

## V Protection and Security

- (5%)(a) Discuss the strengths and weaknesses of implementing an access matrix using *capabilities* that are associated with domains.
- (5%)(b) In the MULTICS system, the protection domains are organized hierarchically into a ring structure. Each ring corresponds to a single domain. The rings are numbered from 0 to 7. Let  $D_i$  and  $D_j$  be any two domains rings. If  $j < i$ , then  $D_i$  is a subset of  $D_j$ . That is, a process executing in domain  $D_j$  has more privileges than does a process executing in domain  $D_i$ . Please describe how UNIX system simplifies this MULTICS design philosophy used for access control protection.
- (5%)(c) In the UNIX operating system, a domain is associated with the user. Switching domains corresponds to changing the user identification temporarily. Please describe how UNIX *setuid bit* provides domain switching from low privilege end user domain to high privilege system program domain.

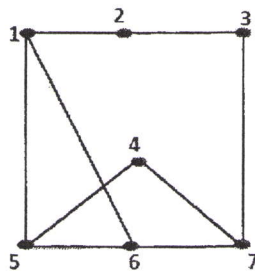
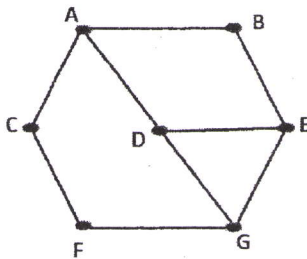
## VI Distributed Systems

- (5%)(a) What is the difference between computation migration and process migration? Which one is easier to implement, why?
- (5%)(b) Please compare and contrast the essential properties of cache-update policy between *write-through* policy and *delayed-write* policy in distributed file system (DFS)?
- (5%)(c) In distributed file system (DFS), cache-consistency problem is the major drawback when we cache data from a file server to clients. How do we verify the validity of cached data of clients and make sure that the cached data in each client is an up-to-date copy?

|         |       |     |       |         |                   |
|---------|-------|-----|-------|---------|-------------------|
| 考 試 科 目 | 計算機數學 | 系 別 | 資訊科學系 | 考 試 時 間 | 2 月 24 日(日) 第 三 節 |
|---------|-------|-----|-------|---------|-------------------|

I 離散數學(60%)

1. (14%) (a). (4%) Suppose that  $a$  and  $b$  are integers,  $a \equiv 11 \pmod{19}$ , and  $b \equiv 3 \pmod{19}$ . Find the largest integer  $c$  with  $0 \leq c \leq 50$  such that  $c \equiv 2a^2 + 3b^2 \pmod{19}$ .  
 (b). (5%) Find the least positive integer of the multiplicative inverse of 101 modulo 4620.  
 (c). (5%) Find the least positive integer  $x$  satisfying the congruence:  $209^{74} \pmod{53}$ .
  
2. (10%) (a). (4%) Let  $R$  be the relation represented by the matrix  $\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}$ . Find the transitive closure of  $R$ .  
 (b). (6%) Find the reflexive closure and the symmetric closure of  $R$  if  $R = \{(1,2), (1,4), (2,3), (3,1), (4,2)\}$ .
  
3. (a). (6%) Find the solution to the recurrence relation  $a_n = 9a_{n-1} - 26a_{n-2} + 24a_{n-3}$  with the initial conditions  $a_0 = 5, a_1 = 19, a_2 = 79$ .  
 (b). (8%) Let  $f(x) = 1/(1-x)^2$ . Find the coefficients  $a_0, a_1, a_2, \dots$  in the expansion  $f(x) = \sum_{k=0}^{\infty} a_k x^k$ .
  
4. (6%) A cubic graph is a simple graph and has every vertex of degree 3.  
 (a). (3%) Draw a cubic graph with 11 vertices, or else prove that there are none.  
 (b). (3%) Draw a cubic graph with 10 edges, or else prove that there are none.
  
5. (8%) Prove or disprove that these two graphs are isomorphic.



|      |       |    |       |      |                 |
|------|-------|----|-------|------|-----------------|
| 考試科目 | 計算機數學 | 系別 | 資訊科學系 | 考試時間 | 2 月 24 日(日) 第三節 |
|------|-------|----|-------|------|-----------------|

6. (8%) A hash function  $H$  with  $H(k)=h$  is a many-to-one function which maps the input  $k$  to the output  $h$ . In addition,  $k$  is from a large set of variable length, called keys, and  $h$  is from a smaller set of a fixed length. A collision of a hash function means that two different keys are mapping to the same output (ie.,  $H(k_i)=h=H(k_j)$  with  $k_i \neq k_j$ ). A good hash function yields few collisions. Assume there are  $n$  keys and  $m$  is the number of possible outputs of a hash function. What is the probability that there is at least one collision occurred in the hash function? In other words, at least two keys are mapped to the same output. Use  $n$  and  $m$  to express this probability.

II. 線性代數(40%) 請書寫必要的計算過程

(10%) 7. Find the value of  $k$  that satisfies the following equation.

$$\det \begin{bmatrix} 2a_1 & 2a_2 & 2a_3 \\ 3b_1 + 5c_1 & 3b_2 + 5c_2 & 3b_3 + 5c_3 \\ 7c_1 & 7c_2 & 7c_3 \end{bmatrix} = k \det \begin{bmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{bmatrix}$$

(10%) 8. Find all eigenvalues and corresponding eigenvectors of following matrix.

(a)  $\begin{bmatrix} 1 & 2 \\ 3 & 2 \end{bmatrix}$  (b)  $\begin{bmatrix} 1 & 1 & 0 \\ 0 & 2 & 2 \\ 0 & 0 & 3 \end{bmatrix}$

(10%) 9. Find the LU factorization of  $A$ , and solve  $x$  in  $Ax=b$

$$A = \begin{bmatrix} 3 & -6 & -3 \\ 2 & 0 & 6 \\ -4 & 7 & 4 \end{bmatrix}, \quad b = \begin{bmatrix} -6 \\ 12 \\ 6 \end{bmatrix}$$

(10%) 10. Let  $V = M_{2 \times 2}(F)$ ,  $W_1 = \left\{ \begin{pmatrix} a & b \\ c & a \end{pmatrix} \in V : a, b, c \in F \right\}$ ,  $W_2 = \left\{ \begin{pmatrix} 0 & a \\ -a & b \end{pmatrix} \in V : a, b \in F \right\}$ .

Find the dimensions of  $W_1$ ,  $W_2$ ,  $W_1 + W_2$ , and  $W_1 \cap W_2$